



**Escuela de
Ingeniería y Arquitectura**
Universidad Zaragoza

Trabajo Fin de Grado

**Clasificación de Potenciales de Error
en una Interfaz Cerebro-Ordenador
en tareas de monitorización**

Carlos Iván Pinós Santamaría

Director: Jason Omedes LLorente

Grado en Ingeniería Informática

Departamento de Informática e Ingeniería de Sistemas
Escuela de Ingeniería y Arquitectura
Universidad de Zaragoza

Septiembre de 2015

Clasificación de Potenciales de Error en una Interfaz Cerebro-Ordenador en tareas de monitorización.

RESUMEN

Las interfaces cerebro-computador(BCI) son sistemas que han experimentado grandes avances en estos últimos años y que tienen aplicaciones en el campo de la rehabilitación o el del entretenimiento. Estas interfaces utilizan la señal cerebral del usuario para controlar diferentes dispositivos, lo que requiere una gran fiabilidad en la detección de los eventos producidos por el cerebro. Sin embargo, esta señal varía dependiendo del usuario y de la tarea lo que provoca que sea difícil detectar los eventos correctamente.

En la literatura, han surgido muchas técnicas para lograr la detección de los eventos pero no se ha obtenido ninguna que se pueda considerar la mejor, aunque si se ha obtenido un proceso común en la detección, que cuenta con un preprocesado de la señal, una extracción de características, y una clasificación con estas características. Las características extraídas suelen estar en el dominio temporal, aunque también se han realizado trabajos en los que se emplean características en otros dominios, como el dominio frecuencial, obteniendo buenos resultados. Tampoco existe un clasificador que obtenga siempre los mejores resultados, aunque se suelen utilizar clasificadores que trabajen bien con pocos datos en comparación al número de características.

Uno de los problemas de la clasificación, es que se suele contar con pocos datos de un sujeto para entrenar el modelo que detecte los eventos. Para lidiar con este problema se puede realizar una clasificación entre sujetos, utilizando los datos de varios sujetos en el entrenamiento, y de otros diferentes para probar el modelo, sin embargo, los resultados suelen deteriorarse al hacer esto. Por ello, han surgido técnicas para esta clasificación entre sujetos, que proyectan las características en otro espacio en el que este problema se minimiza, aunque no se ha conseguido eliminar completamente.

En este contexto, desde la Universidad de Zaragoza se ha trabajado utilizando potenciales de error, un tipo de potencial evocado que surge cuando el usuario detecta un evento inesperado, para proporcionar retro-alimentación durante el uso de las BCIs. Para el análisis de estos potenciales han estado utilizando una metodología de clasificación estándar para detectar los eventos. Esta metodología tiene unos resultados aceptables cuando se clasifica a nivel de sujeto, pero al clasificar entre sujetos, los resultados se degradan.

El objetivo de este trabajo es realizar una comparativa entre esta metodología, utilizando características de diferentes dominios, una metodología con características de fase y una metodología extraída de un concurso de clasificación que hace uso de la Geometría Riemanniana, que tiene ciertas propiedades que permiten generalizar entre sujetos sin que los resultados se vean afectados. Para ello, se han utilizado los datos obtenidos de un experimento en el que 12 sujetos realizaban una tarea de monitorización, del que se han extraído dos conjuntos de datos que han sido utilizados en la clasificación.

Índice

1. Introducción	10
1.1. Motivación	10
1.2. Estructura	12
2. EEG y Potenciales de Error (ErrP)	14
3. Clasificación de errores	17
3.1. Adquisición de la señal	17
3.2. Preprocesado de la señal	17
3.3. Extracción de características	18
3.3.1. Metodología 1: Universidad de Zaragoza	18
3.3.2. Metodología 2: Concurso	19
3.3.3. Metodología 3: Características de Fase	24
3.4. Clasificación	24
3.4.1. Análisis Discriminante Lineal (LDA)	24
3.4.2. Máquinas de vectores de soporte (SVM)	25
3.4.3. Red elástica	25
4. Métodos de clasificación	28
4.1. El experimento	28
4.2. Clasificación	29
4.3. Métricas de comparación	32
4.4. Herramientas e implementación	32
5. Resultados de Clasificación	36
5.1. Clasificación por sujeto	36
5.2. Clasificación por transferencia entre sujetos	38
6. Conclusiones	44
Anexos	49

A. Resultados adicionales	49
A.1. Resultados LDA de un solo sujeto	49
A.2. Resultados Red Elástica de un solo sujeto	50
A.3. Resultados LDA por transferencia entre sujetos	51
A.4. Resultados Red Elástica por transferencia entre sujetos	51

Índice de figuras

2.1.	Se muestra la señal promedio para los eventos de error, los eventos correctos y la diferencia entre ambos. En la parte inferior se muestra el test de discriminabilidad r^2 entre las dos clases de eventos, donde los colores más oscuros indican una mayor diferencia. En la parte superior se muestra la representación del mapa topográfico de la cabeza para los picos más relevantes.	15
3.1.	Distribución de los canales de EEG utilizados en el experimento. Los electrodos de color azul corresponden a la zona fronto-central del cerebro.	18
3.2.	A la izquierda se observa como es la señal con ruido obtenida de la matriz X y a la derecha se muestra como es la señal sin ruido representada por la matriz DA para dos eventos.	21
3.3.	Subespacio Riemanniano M y su espacio tangente local en C. Mediante el mapeo logarítmico $\log_c(C_i)$ se proyecta la matriz de covarianza C_i al espacio tangente. La operación inversa es el mapeo exponencial $\exp_c(P_i)$, siendo P_i un elemento en el espacio tangente.	22
3.4.	En la imagen se puede apreciar como de distintas pueden ser las distancias entre 2 matrices en un subespacio Riemanniano. La línea roja representa la distancia euclídea y la línea verde la distancia Riemanniana en el subespacio.	23
3.5.	A la izquierda se ve como actúa el algoritmo SVM. Las líneas discontinuas representan los vectores de soporte de cada clase, y la línea azul entre ellas el hiperplano que separa las clases. En el centro se observa el algoritmo LDA, siendo los 2 puntos negros las medias de cada clase, y las elipses las matrices de covarianza respectivas. La línea de separación se obtiene como la línea perpendicular a la línea que une los 2 puntos medios. A la derecha se observa como actúa la red elástica que obtiene la línea de separación mediante una regresión.	25
4.1.	Cada repetición del experimento consistía en un círculo amarillo que iba desde una posición inicial a un objetivo concreto (círculo verde oscuro). Cada repetición podía ir a una posición errónea (a) o correcta (c) desde el principio. También existía la posibilidad de que se desviara de su trayectoria provocando un desvío a error (d), en el caso de que comenzara siguiendo la trayectoria correcta, o desvío a correcto(b), en el caso contrario	29

4.2.	A la izquierda aparecen la señal promedio para los desvíos de la clase error y de la clase correcto, y a la derecha, aparecen la señal promedio para los inicios de la clase error y de la clase correcto. La línea azul es la señal promedio para los eventos de error y la roja es la señal promedio para los eventos de no error.	30
4.3.	Se muestra el proceso completo de clasificación en el que se observa el preprocesado de la señal, la extracción de características y los clasificadores utilizados. Las cajas de color azul indican los algoritmos que han sido implementados y las de color naranja, los algoritmos que han sido utilizados con la ayuda de librerías externas.	31
4.4.	Se muestra un ejemplo de curva ROC de la que se obtiene el AUC.	32
5.1.	Se muestran los resultados de clasificación medios de cada metodología por clasificador.	37
5.2.	Se muestran los resultados de cada sujeto por metodología utilizando el clasificador SVM para la clasificación.	39
5.3.	Se muestran los resultados de la clasificación por transferencia entre sujetos.	40

1. Introducción

1.1. Motivación

Las interfaces cerebro-computador (BCI), son sistemas que utilizan la actividad neuronal grabada del cerebro, para realizar acciones mediante dispositivos externos. La señal procesada es utilizada por estos dispositivos para realizar la acción o acciones que corresponda en cada caso, como controlar un brazo robótico, o mover una silla de ruedas [1]. Ya que las BCI no requieren de actividad muscular, estas fueron concebidas en un principio como un sistema de ayuda para pacientes con graves problemas neuromusculares como lesiones medulares o infarto cerebral. Dependiendo en la técnica utilizada para extraer la señal cerebral existen dos principales tipos de BCI, invasivas y no invasivas. Dentro de las no invasivas, la más utilizada es el electroencefalograma (EEG) debido a su buena resolución temporal, su relativo bajo coste y facilidad de uso.

Mediante el uso del electroencefalograma es posible obtener la respuesta del cerebro a varios tipos de respuestas cerebrales, entre los que destacan señales asíncronas como la activación coordinada de grupos de neuronas en ciertas bandas frecuenciales, como se da en el caso de la Imaginación Motora [2], o señales síncronas en respuesta a un estímulo, habitualmente externo, dando lugar a potenciales evocados (EP) o potenciales relacionados a eventos (ERP). Dentro de los potenciales relacionados a eventos, existen los potenciales de error (ErrP) [3], que surgen cuando un sujeto comete o percibe una acción diferente a la esperada o errónea.

Recientemente, se ha propuesto el uso de estos potenciales como una señal natural que permite al usuario dar información de refuerzo para el aprendizaje de dispositivos [4]. Para que el control de estas interfaces sea eficaz y sencillo para el usuario, es importante conseguir detectar con fiabilidad la aparición de dichos potenciales. Sin embargo, esto es complicado porque el EEG contiene un alto nivel de ruido que hace difícil discriminar la señal de interés. Además, la señal cerebral es altamente no estacionaria y varía dependiendo de diversos factores como son el sujeto o la tarea a realizar, por lo que es importante diseñar algoritmos que mejoren su detección. La mayoría de los algoritmos propuestos se componen de un preprocesado inicial de la señal, que tiene por objetivo incrementar el ratio señal-ruido, una metodología de extracción de características para seleccionar las componentes de la señal que permitan detectar mejor el evento, y finalmente, el entrenamiento de un clasificador que aprende a detectar automáticamente cuando ocurre un evento.

Debido al creciente interés que ha habido en el campo de las BCI, se han publicado

diversos resultados relacionados a la investigación y evaluación de estos procesos. Aunque no se ha encontrado ninguno que obtenga siempre los mejores resultados, si que existen técnicas que son comúnmente utilizadas en el procesado de la señal, como son el filtrado espacial, temporal y frecuencial con el objetivo de usar conocimientos previos sobre la señal a analizar y eliminar componentes indeseadas de la señal, o el submuestreo. La extracción de características es más variada y se puede llevar a cabo en diferentes dominios, como el dominio temporal o el de frecuencia [5]. También es habitual la reducción de la dimensionalidad de los datos mediante distintos algoritmos [6] [7] que seleccionen los canales o características más relevantes, y finalmente la normalización de las características obtenidas previas al entrenamiento del clasificador. Además, algunas de estas técnicas requieren la elección de ciertos parámetros para optimizar su funcionamiento, en estos casos se suelen utilizar algoritmos de validación cruzada para seleccionar el valor de dichos parámetros.

A su vez, en la literatura existen discrepancias sobre que clasificador obtiene los mejores resultados. Los clasificadores más populares son los clasificadores lineales, siendo el SVM-lineal y el LDA [8] los más utilizados, aunque también se han empleado algoritmos no lineales como SVM con kernel gaussianos [9], redes neuronales [10], o combinaciones de más de un clasificador [11].

Para medir la eficacia de estos procesos de clasificación, se realizan experimentos en los que se graba la señal de un conjunto de sujetos mientras realizan una serie de tareas en las que se presentan eventos discretos diseñados para evocar los potenciales deseados en la actividad cerebral de los sujetos. Dicha señal se suele segmentar en pequeñas instancias temporales alineadas con la aparición de los eventos. En el caso de los potenciales de error, los eventos presentados se corresponden con acciones erróneas o acciones correctas. Estas instancias temporales se dividen en un subconjunto de entrenamiento en el que se conoce a que clase pertenece cada instancia y permite entrenar un modelo; y un conjunto de validación, en el que hay que predecir la clase de cada instancia y se evalúa la fiabilidad del modelo entrenado. Habitualmente, la clasificación se realiza utilizando los datos de cada sujeto de manera individual, pero en determinadas ocasiones, los datos disponibles obtenidos de sujetos individuales son insuficientes y se han propuesto métodos de clasificación que utilicen transferencia de información entre sujetos (i.e. entrenar usando la señal de un grupo de sujetos y evaluar en un grupo de sujetos diferente). Sin embargo, la señal cerebral es altamente no estacionaria y muy diferente entre sujetos por lo que los resultados decaen bastante.

En este contexto, recientemente se lanzó un concurso público [12] cuyo objetivo era precisamente proponer un método de clasificación que combinase los datos de la señal cerebral grabada de varios sujetos para detectar potenciales de error en la señal procedente de otros sujetos realizando una tarea de detección [13]. Los ganadores de dicho concurso proponen una metodología que hace uso de matrices de covarianza y de la geometría Riemanniana para detectar los potenciales de error en este escenario intra-sujetos.

El objetivo de este trabajo es realizar una comparativa entre los métodos de procesado y extracción de características utilizados en experimentos de potenciales de error desarrollados en la Universidad de Zaragoza, un método que hace uso de las

características fase, que han sido poco utilizadas en la literatura pero que han sido capaces de producir buenos resultados, y el método propuesto por los ganadores del concurso. Para ello, se analizarán tanto los resultados obtenidos de clasificar la señal de los sujetos individualmente como los resultados de hacer transferencia entre sujetos. Este análisis se ha llevado a cabo utilizando los datos obtenidos de un experimento realizado en la Universidad de Zaragoza, en el que 12 voluntarios realizaron una tarea de monitorización, donde un dispositivo virtual ejecutaba ciertas trayectorias predefinidas y los usuarios tenían que evaluar mentalmente dichos movimientos como correctos o erróneos.

Para realizar esta comparativa, se ha dividido el trabajo en las siguientes tareas:

1. Estudio del estado del arte para la clasificación de BCIs y, en particular, para la clasificación de potenciales de error.
2. Diseño de la metodología de comparación (selección de conjuntos de datos y métodos).
3. Búsqueda e integración de algoritmos en un entorno común para realizar la comparativa.
4. Desarrollo de un nuevo método basado en características de fase
5. Obtención y comparación de los resultados con las tres metodologías seleccionadas, la de la universidad, la del concurso y la que utiliza características de fase.

1.2. Estructura

Este trabajo está organizado en cinco partes que se detallan a continuación:

- **Capítulo 2: EEG y Potenciales de Error:** En este capítulo se describen de manera más exhaustiva tanto la señal de EEG como los potenciales de error (ErrP).
- **Capítulo 3: Clasificación de errores:** En este capítulo se explican las diferentes técnicas de procesado y clasificadores empleados.
- **Capítulo 4: Métodos de clasificación:** En este capítulo se explica el experimento realizado y se muestran los conjuntos de datos extraídos del mismo. Además, se indican las técnicas que componen cada metodología y las métricas de evaluación que se han utilizado para su comparación.
- **Capítulo 5: Resultados:** En este capítulo se exponen los distintos resultados obtenidos y se realiza un análisis de los mismos.
- **Capítulo 6: Conclusiones:** Por último, se exponen las conclusiones obtenidas del trabajo.

2. EEG y Potenciales de Error (ErrP)

El electroencefalograma (EEG) es un sistema de monitorización que graba la actividad eléctrica producida por el cerebro. Es una técnica no invasiva, es decir, que no requiere el implante de ningún electrodo en la corteza cerebral. El equipo necesario para grabar la señal, consta de un casco con electrodos que se coloca en la zona superior de la cabeza del sujeto y un equipo que registra las fluctuaciones del potencial eléctrico captadas por los electrodos. Entre los métodos no invasivos, éste es el más utilizado debido a su relativo bajo coste y buena resolución temporal [14]. De entre los diferentes patrones que se pueden extraer de esta señal, destaca un tipo de potenciales evocados denominados potenciales de error (ErrP), que son el objetivo de este estudio. Los potenciales de error muestran la modificación del potencial eléctrico producido por el sistema nervioso cuando un sujeto comete un error o cuando éste observa una conducta errónea o diferente a la esperada.

El análisis de estos potenciales se suele llevar acabo en tareas discretas, donde hay un estímulo conocido. Dependiendo de la tarea, se han observado diferentes tipos de potenciales de error. Los errores de respuesta, que surgen cuando el sujeto comete un error durante tareas en las que se requiere ejecutar reacciones lo más rápido posible [15]; los errores de observación, que surgen cuando el sujeto observa a otro sujeto cometer un error [16]; y los errores de interacción, que aparecen cuando el sujeto observa a una máquina realizar una acción diferente a la acción esperada [17], siendo estos últimos en los que se centra este trabajo.

En el contexto de la detección automática de dichos potenciales de error, se ha conseguido entrenar clasificadores que aprenden características relevantes de la señal cerebral, siendo capaces de posteriormente diferenciar cuando un cierto sujeto está presenciando un evento erróneo o no. Para llevar a cabo este proceso de aprendizaje, es habitual seleccionar una ventana centrada en el momento en el que ocurre un evento al que se le asigna una etiqueta dependiendo si es correcto o erróneo. Posteriormente se aplica un proceso de filtrado y se extraen las características de la señal que ayuden a discriminar mejor entre ambas clases.

Las instancias individuales de la señal obtenida para dichos eventos están altamente contaminadas por ruido procedente de la señal cerebral de fondo, por lo que a simple vista es casi imposible percibir las diferencias entre las diferentes clases. Para ello, a la hora de hacer estudios neurológicos de la señal, una técnica común es hacer el promedio de las diferentes instancias de cada evento con el objetivo de incrementar el ratio señal-ruido y así observar los patrones más relevantes que dominan el proceso. En la Figura 2.1 se

2EEG y Potenciales de Error (ErrP)

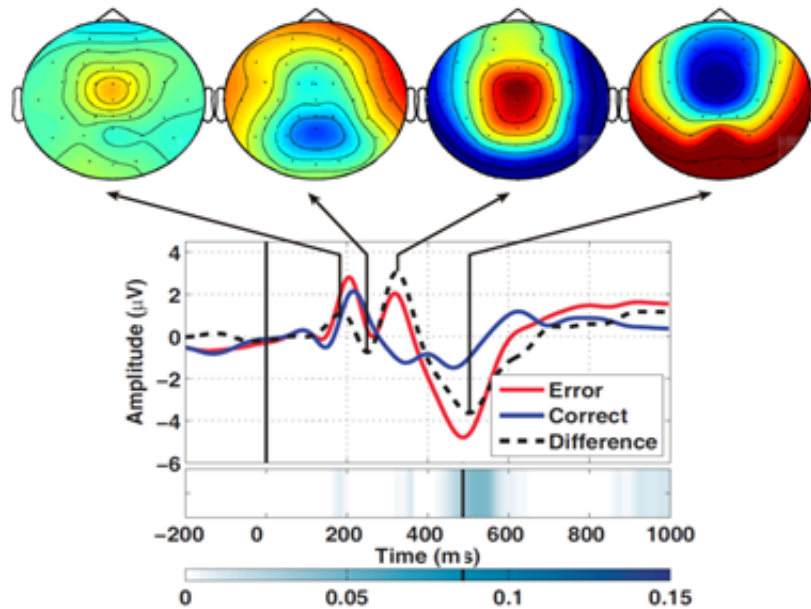


Figura 2.1: Se muestra la señal promedio para los eventos de error, los eventos correctos y la diferencia entre ambos. En la parte inferior se muestra el test de discriminabilidad r^2 entre las dos clases de eventos, donde los colores más oscuros indican una mayor diferencia. En la parte superior se muestra la representación del mapa topográfico de la cabeza para los picos más relevantes.

muestra una comparación entre los promedios de los eventos de error y los de los eventos correctos siendo la señal denominada como potencial de error la diferencia entre ambas (línea discontinua). Estos potenciales están caracterizados por un pico positivo a los 200 ms tras la aparición del error y un pico negativo a los 250 ms, seguidos de una modulación positiva de mayor amplitud aproximadamente a los 300 ms y una deflacción más amplia alrededor de los 450 ms [4], aunque dichas latencias pueden variar según el experimento y el sujeto. Además, hay evidencias que sugieren que los errores están originados en el córtex del cíngulo anterior y por lo tanto su actividad se ve mejor reflejada en los canales de las áreas fronto-centrales del cerebro (FCz, Cz) [18].

3. Clasificación de errores

La principal tarea de las BCI consiste en ser capaz de decodificar automáticamente los comandos mentales aportados por un usuario a partir de su señal cerebral. Para ello, es habitual fragmentar dicha señal en distintos segmentos de tiempo denominados trials (a falta de una mejor palabra para definirlos en español), que pertenecen a una clase Z . En el caso de la clasificación de los potenciales de error $Z = 1,2$, evento correcto o evento erróneo respectivamente. Dichos trials se separan en un conjunto de entrenamiento de los cuales su etiqueta es conocida, por lo que se emplean para entrenar un modelo; y el conjunto de testeo de cuyos trials se supone que su etiqueta es desconocida y es el modelo el que tiene que averiguar a que clase pertenece. Debido a que el EEG tiene un ratio señal ruido muy bajo antes de entrenar el modelo se suelen aplicar procesos que mejoren la discriminabilidad entre las clases a clasificar. Entre estos procesos cabe destacar la elección de características utilizadas para entrenar el modelo. En este capítulo se presentan 3 metodologías de extracción de características distintas. Una utilizada habitualmente en experimentos realizados por la Universidad de Zaragoza, otra metodología que hace uso de características de fase y otra ganadora de un reciente concurso de detección de errores. Adicionalmente, los resultados obtenidos de ambas metodologías se van a testear en 3 clasificadores diferentes, LDA, SVM y Red Elástica, que son los más utilizados en la literatura, pero no está claro cual reporta mejores rendimientos.

3.1. Adquisición de la señal

La actividad electroencefalográfica (EEG) fue grabada utilizando un sistema g.Tec g.USBamp que utilizaba 60 electrodos colocados de acuerdo al sistema internacional extendido 10/20 con la toma de tierra en FPz y la referencia en el lóbulo de la oreja izquierda, como se observa en la figura 3.1.

La señal se digitalizó con una frecuencia de muestreo de 256 Hz y con un filtrado Notch a los 50 Hz.

3.2. Preprocesado de la señal

En primer lugar, a la señal grabada se le aplicó un filtro de paso banda Butterworth de fase zero de cuarto orden entre 1 y 10 Hz. Después, fue filtrada mediante un filtro

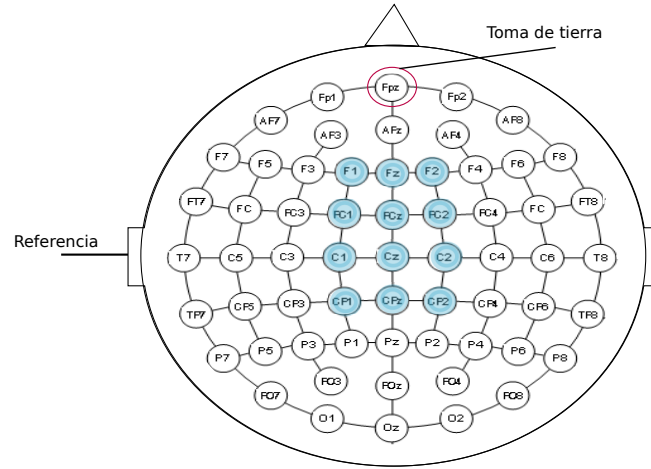


Figura 3.1: Distribución de los canales de EEG utilizados en el experimento. Los electrodos de color azul corresponden a la zona fronto-central del cerebro.

espacial llamado CAR, del inglés, Common Average Reference [22]. Este filtro sustrae la media de todos los canales por instante temporal, eliminando parte de la señal de fondo común a todos los electrodos.

3.3. Extracción de características

Se va a definir como trial a la matriz $X \in \mathbb{R}^{N_c \times N_t}$, en la que N_c es el número de canales utilizados y N_t la duración de la ventana temporal que contiene la información relevante de cada evento. En este caso, el número de canales N_c utilizados es 60 canales, y la ventana temporal es de un segundo desde la aparición del evento, lo que supone 256 muestras por canal. Sobre dichos trials se aplicarán separadamente las tres metodologías de extracción de características definidas a continuación:

3.3.1. Metodología 1: Universidad de Zaragoza

En esta metodología se han utilizado dos tipos de características: temporales y frecuenciales.

Las características temporales son los voltajes de cada trial tomando los 12 canales fronto-centrales (FC1, FCz, FC2, C1, Cz, C2, CP1, CPz, CP2, P1, Pz, P2), con una ventana temporal de un segundo desde la aparición del evento. Dado que la señal se digitalizó con una frecuencia de muestreo de 256 Hz, se obtienen 256 características temporales por canal, que son submuestreadas a 32 Hz, pasando a 32 componentes. A continuación se concatenan las características de cada canal, dando como resultado un vector de 384 características, que es normalizado utilizando una normalización estándar de media 0 y desviación estándar 1.

Para la extracción de las características frecuenciales se estima la potencia espectral

de cada uno de los canales utilizados anteriormente, y para la misma ventana temporal de un segundo, mediante el método de Welch [23]. Este método divide la ventana temporal en varios segmentos con cierta superposición entre ellos, calcula el periodograma de cada segmento y los promedia, obteniendo un resultado más robusto que aplicando la transformada de Fourier directamente. Las características frecuenciales son los valores de potencia obtenida en la banda theta de frecuencia ($[4-8] \pm 1$ Hz), dado que en la bibliografía se ha reportado que es la banda en la que se manifiestan los potenciales de error [24]. De esta manera se obtienen 7 características por canal, que se concatenan obteniendo un total de 84 características a las que se les aplica una normalización z-score.

3.3.2. Metodología 2: Concurso

Este método propone utilizar las características del espacio Riemanniano, que permiten obtener un modelo con una mejor capacidad de generalización intra-sujeto, respecto a otros espacios. Para ello, se hace uso de las matrices de covarianza, que son simétricas y definidas positivas (SPD), cuyo espacio forma un subespacio Riemanniano [21]. Además, estas matrices tienen ciertas propiedades entre las que destaca la propiedad de la invarianza ante transformaciones congruentes, que permite aplicar transformaciones lineales sin que su distancia en dicho subespacio sea afectada.

El calculo de cada matriz de covarianza se realiza con la siguiente ecuación:

$$Cov = \frac{X X^T}{N_t - 1} \quad (3.1)$$

Siendo X un trial de la forma $X \in \mathbb{R}^{N_c \times N_t}$ y T es la operación traspuesta de una matriz. Estas matrices de covarianza no resultan eficientes para la clasificación de ERPs, ya que la amplitud de la señal de interés es del mismo orden de magnitud que la señal de fondo del EEG. Por lo tanto, la información espacial contenida en esta matriz no es suficiente para la clasificación. Para conseguir una matriz de covarianza con tanto la información temporal como la espacial, se construye lo que vamos a denominar como super-trial X_s , añadiendo el patrón promedio de los trials de las clases a clasificar:

$$X_s = \begin{pmatrix} M_c \\ M_e \\ X \end{pmatrix} \quad (3.2)$$

Donde M_c es la matriz promedio para los trials de la clase correcta y M_e la matriz promedio para los trials de la clase de error. Calculando la matriz de covarianza del super-trial obtenemos una matriz con la siguiente estructura:

$$C_s = \begin{pmatrix} M_c(M_c)^T & M_c(M_e)^T & M_c(X)^T \\ M_e(M_c)^T & M_e(M_e)^T & M_e(X)^T \\ X(M_c)^T & X(M_e)^T & X(X)^T \end{pmatrix} \quad (3.3)$$

Nótese que los elementos compuestos por M_c y M_e son comunes para los trials y no son relevantes para la clasificación. El elemento $X(X)^T$ contiene la información espacial.

El resto de elementos muestran la correlación de cada trial con cada clase. El elemento $M_c(X)^T$ contendrá un valor elevado en los trials que correspondan a la clase correcta, y el elemento $M_e(X)^T$ contendrá un valor elevado para los trials de la clase error. De esta manera se ha conseguido que las matrices de covarianza contengan la información temporal y la espacial de los datos.

Adicionalmente, una medida habitual en estos métodos es utilizar un filtro espacial con el objetivo de realzar la señal de interés. Para ello, se va a utilizar el algoritmo XDAWN [6], cuya meta es maximizar la señal respecto a la señal más ruido. Este algoritmo considera que la señal, es la suma de un potencial evocado y ruido:

$$X = DA + N \quad (3.4)$$

En la que X representa la señal, D es la matriz de Toeplitz en la que los elementos de la primera columna son cero, si no hay estímulo, o uno, si el estímulo de interés está presente. A representa la respuesta cerebral al estímulo de interes y N representa el ruido. En la Figura 3.2 puede observarse un ejemplo sintético de la representación de los elementos de la ecuación anterior. De dicha ecuación el valor de la señal deseada puede calcularse mediante mínimos cuadrados:

$$\hat{A} = \arg \min_A \|X - DA\|_2^2 \quad (3.5)$$

A partir de la ecuación 3.4, el algoritmo XDAWN propone encontrar el filtro U que maximice la señal respecto a la señal más ruido mediante la siguiente ecuación:

$$XU = DAU + NU \quad (3.6)$$

Dicho filtro se estima como se observa a continuación:

$$\hat{U} = \arg \max_U \frac{\text{Tr}(U^T \hat{A} D^T D \hat{A} U)}{\text{Tr}(U^T X^T X U)} \quad (3.7)$$

Dónde \hat{U} es la estimación del filtro, \hat{A} la estimación de la señal sin ruido calculada en la ecuación 3.5 y $\text{Tr}(\cdot)$ el operador de traza. De esta manera, aplicando el algoritmo XDAWN a los trials de la clase error obtenemos el filtro espacial U_e , y procediendo de la misma manera para los trials de la clase correcto obtenemos U_c . Posteriormente, se preservan las 5 componentes más significativas de cada filtro y se aplican ordenadamente a las componentes que conforman el super-trial de la ecuación 3.2 obteniendo un super-trial reducido:

$$\begin{pmatrix} M_c U_c \\ M_e U_e \\ X U_c \\ X U_e \end{pmatrix} = \begin{pmatrix} M_{rc} \\ M_{re} \\ X_{rc} \\ X_{re} \end{pmatrix} = X_{rs} \quad (3.8)$$

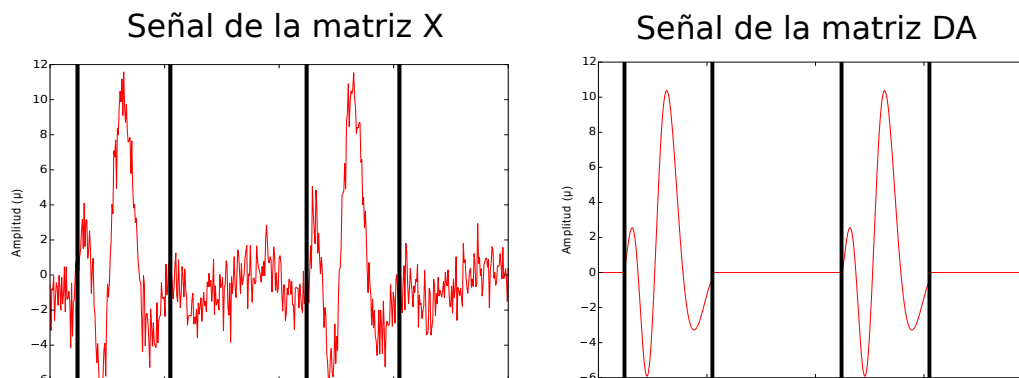


Figura 3.2: A la izquierda se observa como es la señal con ruido obtenida de la matriz X y a la derecha se muestra como es la señal sin ruido representada por la matriz DA para dos eventos.

De esta manera, a parte de realzar la señal, se consigue reducir la dimensión de un super trial de $3N_c \times N_t$ (donde N_c son 60 electrodos para nuestro estudio) a $4 \times 5 \times N_t$ dimensiones, consiguiendo matrices de covarianza reducidas de la forma:

$$C_{rs} = \frac{X_{rs}X_{rs}^T}{N_t - 1} \quad (3.9)$$

A continuación, se realiza un proceso de selección de componentes mediante un algoritmo basado en maximizar la diferencia entre clases por medio de la distancia Riemanniana [25]. Dicho algoritmo calcula un centroide para cada una de las clases, correcto y error (C_c y C_e respectivamente). Estos centroides son el resultado de calcular la media geométrica de las matrices de covarianza de cada clase. Dicha media geométrica se calcula mediante un proceso iterativo, en el que las matrices de covarianza se proyectan en un espacio tangente, en el que la distancia Riemanniana puede aproximarse por la distancia Euclídea. Esta proyección se realiza utilizando el *mapeo logarítmico* de las matrices de covarianza con la siguiente ecuación:

$$P_{rs} = \text{Log}_C(C_{rs}) = C^{1/2} \log_m(C^{-1/2} C_{rs} C^{-1/2}) C^{1/2} \quad (3.10)$$

en el que C es el punto o centroide sobre el que se calcula el espacio tangente al subespacio Riemanniano, \log_m la operación logaritmo de una matriz, y P_{rs} la matriz proyectada en el espacio tangente.

Una vez se han proyectado todas las matrices de covarianza en el espacio tangente, se calcula la media aritmética y se deshace la transformación. Esta operación se conoce como *mapeo exponencial*:

$$C_{rs} = \text{Exp}_C(P_{rs}) = C^{1/2} \text{exp}_m(C^{-1/2} P_{rs} C^{-1/2}) C^{1/2} \quad (3.11)$$

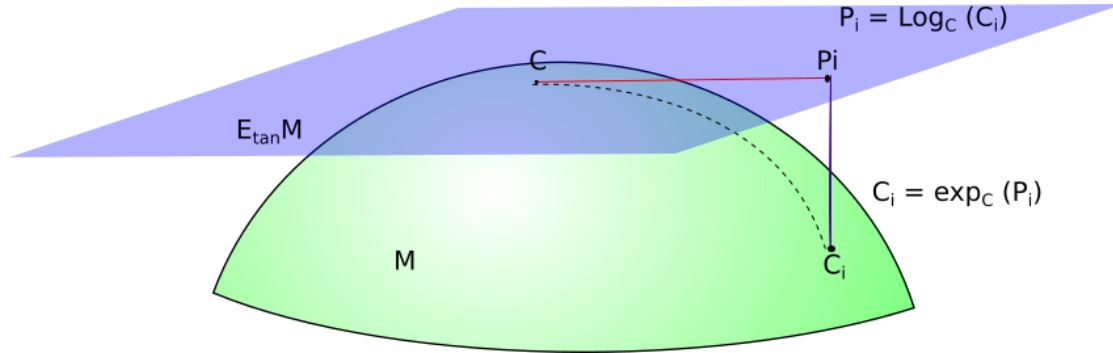


Figura 3.3: Subespacio Riemanniano M y su espacio tangente local en C. Mediante el mapeo logarítmico $\log_C(C_i)$ se proyecta la matriz de covarianza C_i al espacio tangente. La operación inversa es el mapeo exponencial $\exp_C(P_i)$, siendo P_i un elemento en el espacio tangente.

siendo \exp_C la exponencial de una matriz. Este proceso se itera hasta lograr la convergencia como muestra el algoritmo 1 y se repite para obtener los centroides de ambas clases C_c y C_e .

Algoritmo 1 Cálculo de la media geométrica de matrices de covarianza SPDs.

Entrada: N matrices de covarianza C_n y error ϵ

Salida: C es la media geométrica calculada

- 1: $C^{(1)} = \frac{\sum_{n=1}^N C_n}{N}$ {Media inicial}
 - 2: **repetir**
 - 3: $P = \frac{\sum_{n=1}^N \text{Log}_{C^{(i)}}(C_n)}{N}$ {Calculamos la media en el espacio tangencial}
 - 4: $C^{(i+1)} = \text{Exp}_{C^{(i)}}(P)$ {Proyectamos el resultado de vuelta al subespacio Riemanniano}
 - 5: **hasta que** $\|C^{(i)} - C^{(i+1)}\| < \epsilon$
-

Tras obtener los 2 centroides, se procede a seleccionar las componentes mediante un proceso iterativo que elimina una componente en cada iteración hasta quedarse con el número deseado. Nótese que la eliminación de una componente afecta a una fila y una columna de los centroides, por lo que no es necesario volver a calcular la media en cada iteración. El criterio de selección del algoritmo es la distancia Riemanniana entre los centroides C_e y C_c , que se define de la siguiente forma:

$$\text{dist}_{riemann}(C_e, C_c) = \|\text{logm}(C_e^{-1}C_c)\|_F = \sqrt{\sum_{n=1}^{N_c} \text{logm}^2 \lambda_n} \quad (3.12)$$

Siendo los λ_n los valores propios de la matriz $C_e^{-1}C_c$, y $\|\cdot\|_F$ la norma de Frobenius de una matriz. Para más detalles, ver el algoritmo 2.

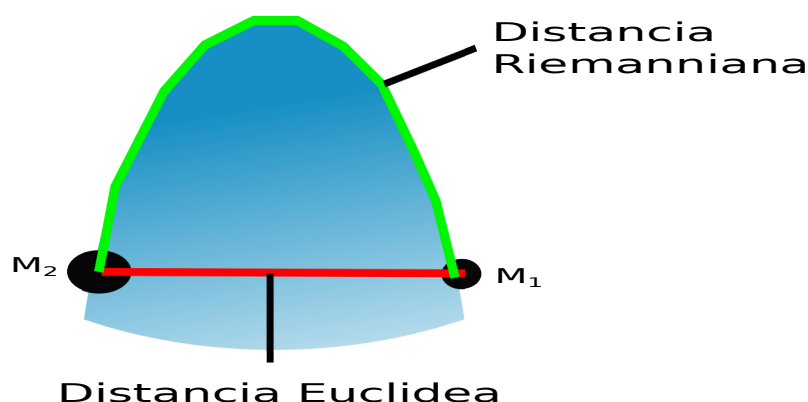


Figura 3.4: En la imagen se puede apreciar como de distintas pueden ser las distancias entre 2 matrices en un subespacio Riemanniano. La línea roja representa la distancia euclídea y la línea verde la distancia Riemanniana en el subespacio.

Algoritmo 2 Selección de componentes mediante distancia Riemanniana.

Entrada: Centroides C^{m1} , C^{m2} y N número de componentes a seleccionar

Salida: Devuelve los componentes seleccionados

- 1: $Componentes = [1..N_c]$ {Empezamos con todos los componentes inicialmente}
 - 2: **para** $i = 1$ a $N_c - N_s$ {Una vez por cada componente a eliminar} **hacer**
 - 3: **para** $j = 1$ a $N_c - j + 1$ {Una vez por cada componente que tenemos actualmente} **hacer**
 - 4: $D_1 = C^{m1} - j$ {Cogemos el centroide actual sin el componente j }
 - 5: $D_2 = C^{m2} - j$
 - 6: $D(j) = dist_{riemann}(D_1, D_2)$ {Calculamos la distancia Riemanniana }
 - 7: **fin para**
 - 8: $max = \arg \max_j D(j)$
 - 9: $C^{m1} = C^{m1} - max$ {eliminamos componente seleccionada del centroide}
 - 10: $C^{m2} = C^{m2} - max$
 - 11: $Componentes = Componentes - max$ {eliminamos componente seleccionada de la lista de componentes}
 - 12: **fin para**
-

Posteriormente se proyectan las matrices de covarianza en un espacio tangente. Este último paso es necesario ya que los clasificadores convencionales trabajan con la distancia Euclídea, que puede ser muy diferente de la distancia Riemanniana que utilizan las matrices de covarianza, como se puede apreciar en la Figura 3.4. Finalmente, dado que los clasificadores convencionales trabajan con vectores en lugar de matrices, se vectorizan las matrices. La vectorización se consigue cogiendo la parte triangular superior de la matriz y multiplicando los elementos que no estaban en la diagonal por $\sqrt{2}$ para conservar la norma.

3.3.3. Metodología 3: Características de Fase

Las características de fase se obtienen aplicando en primer lugar la transformada de Hilbert a los 12 canales fronto-centrales (FC1, FCz, FC2, C1, Cz, C2, CP1, CPz, CP2, P1, Pz, P2) y con una ventana temporal de un segundo:

$$\hat{x}(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau. \quad (3.13)$$

Después se calcula la señal analítica juntando la parte real, x , con la parte imaginaria, \hat{x} :

$$y = x + i\hat{x} \quad (3.14)$$

Dichas características se dividen en *magnitud instantánea*, definida como el valor absoluto o envolvente de la señal:

$$magnitud = |y| \quad (3.15)$$

Y la *fase instantánea*, obtenida de la siguiente expresión:

$$fase = atan2(imag(y), real(y)) \quad (3.16)$$

Dichas características son submuestreadas a 32 Hz, obteniendo 32 de características de fase y 32 de magnitud por canal. Tras concatenar las características de cada canal, se obtiene un vector de 768 características que se normaliza con el método z-score.

3.4. Clasificación

3.4.1. Análisis Discriminante Lineal (LDA)

El LDA [8] es un algoritmo de clasificación comúnmente utilizado que trabaja bien con pocos datos y no requiere elegir ningún parámetro. Este algoritmo asume que los datos siguen una distribución gaussiana y las clases a clasificar comparten la misma matriz de covarianza Σ . El algoritmo LDA busca la proyección lineal ω que mejor separe ambas clases, mediante la siguiente fórmula:

$$\omega = \Sigma^{-1}(\mu_1 - \mu_2) \quad (3.17)$$

siendo μ_1 y μ_2 la media de cada clase y Σ la matriz de covarianza media.

$$\Sigma = \frac{\Sigma_1 + \Sigma_2}{2} \quad (3.18)$$

con Σ_1 y Σ_2 las matrices de covarianza de cada clase. Esta matriz puede ser regularizada en el caso de que esté mal condicionada (pocos datos):

$$\Sigma = (1 - \lambda)\Sigma + \lambda I \quad (3.19)$$

El parámetro de regularización λ se puede calcular con el método de Ledoit-Wolf [26].

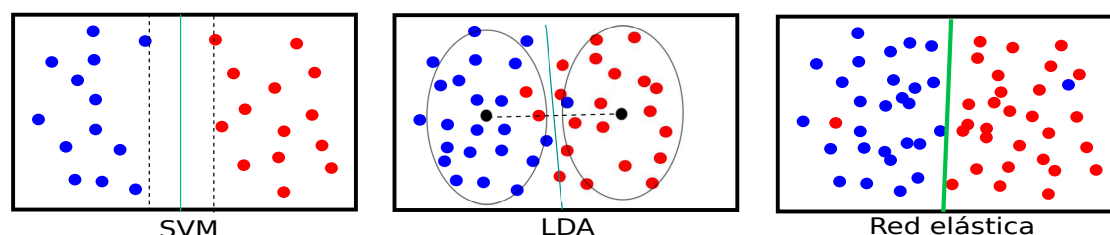


Figura 3.5: A la izquierda se ve como actúa el algoritmo SVM. Las líneas discontinuas representan los vectores de soporte de cada clase, y la línea azul entre ellas el hiperplano que separa las clases. En el centro se observa el algoritmo LDA, siendo los 2 puntos negros las medias de cada clase, y las elipses las matrices de covarianza respectivas. La línea de separación se obtiene como la línea perpendicular a la línea que une los 2 puntos medios. A la derecha se observa como actúa la red elástica que obtiene la línea de separación mediante una regresión.

3.4.2. Máquinas de vectores de soporte (SVM)

El SVM [27] es un algoritmo que ha incrementado su popularidad debido a su capacidad de contemplar relaciones no lineales fácilmente mediante el uso del truco del kernel, siendo uno de los kernels más utilizados el RBF [28]. Este clasificador busca el mejor hiperplano que separe las clases del modelo. A diferencia de otros algoritmos, el SVM no tiene en cuenta todos los ejemplos disponibles, tiene en cuenta únicamente los ejemplos que más cerca estén de una clase a la otra, estos ejemplos se conocen como vectores de soporte. Una vez encontrados, se sitúa el hiperplano en una posición intermedia entre los dos vectores de soporte. Al utilizar el kernel RBF se introducen además 2 parámetros para elegir, γ y c . El parámetro γ sirve para indicar el área de influencia de los vectores de soporte. Con un valor muy bajo, el área de influencia de los vectores de soporte aumenta y se produce subajuste, al no poder capturar bien la complejidad de los datos. Si el valor γ es muy grande, el área de influencia es muy pequeña provocando sobreajuste. El parámetro c sirve para penalizar la complejidad del modelo. Al bajar el valor de c disminuyes la penalización al clasificar mal un ejemplo y al incrementarlo la aumentas. Por esto, si eliges un parámetro c muy bajo puede producirse subajuste mientras que si eliges un parámetro c muy alto se puede producir sobreajuste. Los valores seleccionados han sido de 1 para el parámetro c y $1/n_{caracteristicas}$ para el parámetro γ .

3.4.3. Red elástica

La red elástica [29] es una regresión lineal regularizada que aplica tanto la norma L1 como la norma L2. La función objetivo en este caso es la siguiente:

$$\min_w \frac{1}{2n_{samples}} \|Xw - y\|_2^2 + \delta \|w\|_1 + \frac{(1 - \delta)}{2} \|w\|_2^2 \quad (3.20)$$

El parámetro δ indica la importancia que se da a una norma sobre la otra, por lo que

si δ es 1 se aplica únicamente la norma L1, y si es 0, solo la norma L2. Para la clasificación ha tomado el valor de 0.5, por lo que se da a ambas normas la misma importancia.

4. Métodos de clasificación

En este capítulo se describe el protocolo experimental utilizado en el que se realizó la grabación de la señal electroencefalográfica analizada en este estudio. En este experimento, 12 sujetos realizaron una tarea de monitorización, en el que un dispositivo virtual ejecutaba ciertas trayectorias predefinidas en monitor de ordenador, y los usuarios tenían que visualizar dichos movimientos y evaluarlos mentalmente como correctos o erróneos.

Adicionalmente, se definen las secuencias de procesamiento que se han seguido para evaluar la señal, y las métricas utilizadas para su comparación. Finalmente, se describen las herramientas utilizadas por el autor de este estudio a la hora de implementar los diferentes análisis descritos en esta memoria.

4.1. El experimento

Doce sujetos participaron en el estudio grabado en la Universidad de Zaragoza. Los voluntarios se sentaron a un metro aproximadamente de la pantalla del ordenador donde observaban una tarea 2D. La tarea consistía en un dispositivo digital que se movía a ocho posibles objetivos distribuidos uniformemente sobre un círculo de 20 centímetros de diámetro (ver figura 4.1). Cada recorrido se definía de la siguiente manera. Primero, se iluminaba uno de los ocho posibles destinos, representando el objetivo actual. Después, el dispositivo se movía realizando una de las siguientes trayectorias:

- **Inicio Erróneo (SE):** La pelota se movía siguiendo una trayectoria en línea recta hasta una posición diferente a la deseada, Figura (4.1a).
- **Inicio Erróneo (SE) y desvío correcto (DC):** La pelota empieza igual que en el caso anterior, pero a mitad del recorrido cambia de dirección y llega a la posición esperada, Figura (4.1b).
- **Inicio Correcto (SC):** La pelota se movía siguiendo una trayectoria en línea recta hasta la posición deseada, Figura (4.1c).
- **Inicio Correcto (SC) y desvío error (DE):** El dispositivo empieza igual que en el caso anterior, pero a mitad de recorrido cambia de dirección hasta una posición no deseada, Figura (4.1d).

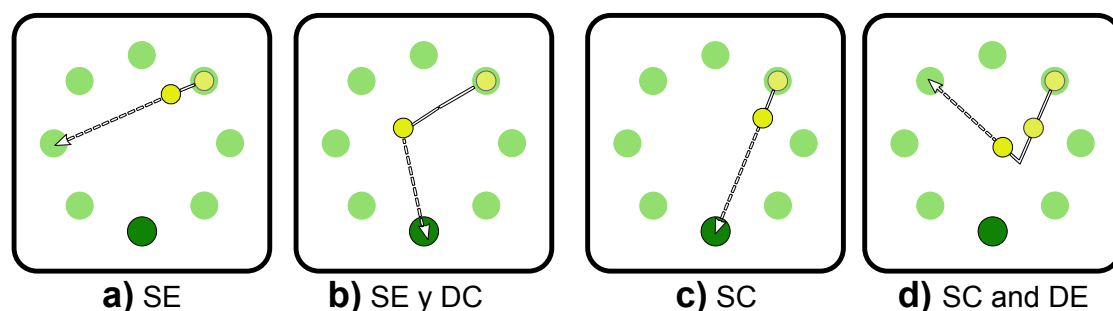


Figura 4.1: Cada repetición del experimento consistía en un círculo amarillo que iba desde una posición inicial a un objetivo concreto (círculo verde oscuro). Cada repetición podía ir a una posición errónea (a) o correcta (c) desde el principio. También existía la posibilidad de que se desviara de su trayectoria provocando un desvío a error (d), en el caso de que comenzara siguiendo la trayectoria correcta, o desvío a correcto(b), en el caso contrario

La probabilidad de que hubiera un inicio correcto o erróneo era la misma (50%) y la probabilidad de desvío dentro de la trayectoria era del 30%. Cada sesión estaba compuesta de 12 bloques con 40 movimientos por bloque entre los que el sujeto descansaba durante unos minutos. La duración total de cada sesión era de unas 2 horas contando la colocación del equipo al sujeto. En total se grabaron 240 trials de inicio correcto, 240 trials de inicio error, de los cuales 75 eran de desvío error y 75 de desvío correcto por sujeto. Los movimientos de cada bloque eran escogidos de manera pseudoaleatoria, y a los sujetos se les instruía que evaluaran mentalmente los recorridos ejecutados por el dispositivo como correctos o erróneos.

En total aparecen cuatro eventos diferentes: Inicio Erróneo, Inicio Correcto, Desvío Erróneo y Desvío Correcto. Estos eventos son los que van a ser utilizados en este trabajo, clasificando los trials de Inicio Erróneo respecto a los de Inicio Correcto, y los de Desvío Erróneo respecto a los Desvío Correcto. En la Figura 4.2 se observa como es la señal promedio de estos eventos.

4.2. Clasificación

Para realizar el análisis de las metodologías, se ha llevado a cabo una clasificación por sujeto y una clasificación por transferencia entre sujetos.

En la clasificación por sujeto se va a diferenciar los eventos de Inicio Correcto respecto a los de Inicio Erróneo, y los eventos Desvío Correcto respecto a los de Desvío Erróneo. Para ello, se van a utilizar 4 secuencias de procesamiento de la señal distintas, que comparten el preprocesamiento explicado en el apartado 3.2, y utilizan los algoritmos SVM, LDA y Red Elástica para clasificar. Las dos primeras secuencias utilizan los métodos empleados por la universidad de Zaragoza, descritos en el apartado 3.3.1, pero con diferentes combinaciones de características. La primera secuencia (Método 1.1) utiliza características temporales para la clasificación, la segunda (Método 1.2)

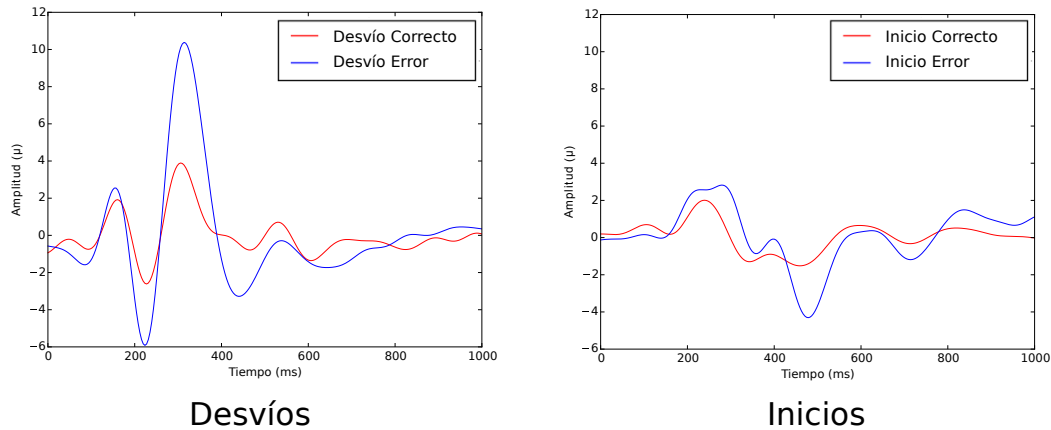


Figura 4.2: A la izquierda aparecen la señal promedio para los desvíos de la clase error y de la clase correcto, y a la derecha, aparecen la señal promedio para los inicios de la clase error y de la clase correcto. La línea azul es la señal promedio para los eventos de error y la roja es la señal promedio para los eventos de no error.

características tanto temporales como de frecuencia. La tercera secuencia (Método 2) utiliza los métodos del concurso explicados en el apartado 3.3.2. Finalmente, la cuarta secuencia utiliza las características de fase (Método 3). Para evaluar los porcentajes de clasificación obtenidos de cada sujeto se utilizó un 5-fold cronológico, en el que los trials de cada sujeto fueron divididos en un conjunto de entrenamiento, compuesto por el 80% de los trials, y en un conjunto de test compuesto por el 20% restante. Nótese, que se realiza un 5-fold cronológico ya que los trials están correlados entre sí debido a la no estacionalidad de la señal. Los valores de normalización de las características fueron obtenidos del conjunto de entrenamiento; así como ciertos parámetros ajustables como la selección del número de componentes o hiperparámetros de los clasificadores que fueron ajustados utilizando un proceso de validación cruzada.

En la clasificación por transferencia se van a diferenciar los eventos de Inicio Correcto respecto a los de Inicio Erróneo, y los eventos Desvío Correcto respecto a los de Desvío Erróneo, utilizando las mismas secuencias de procesamiento que en la clasificación por sujeto. Para evaluar los porcentajes de clasificación obtenidos entre sujetos se utilizó un 5-fold, en el que los trials se dividieron en un conjunto de entrenamiento, compuesto por los trials de 9 de los sujetos, y en un conjunto de test compuesto por los trials de los 3 sujetos restantes. Los valores de normalización de las características fueron obtenidos del conjunto de entrenamiento; así como ciertos parámetros ajustables como la selección del número de componentes o hiperparámetros de los clasificadores que fueron ajustados utilizando un proceso de validación cruzada.

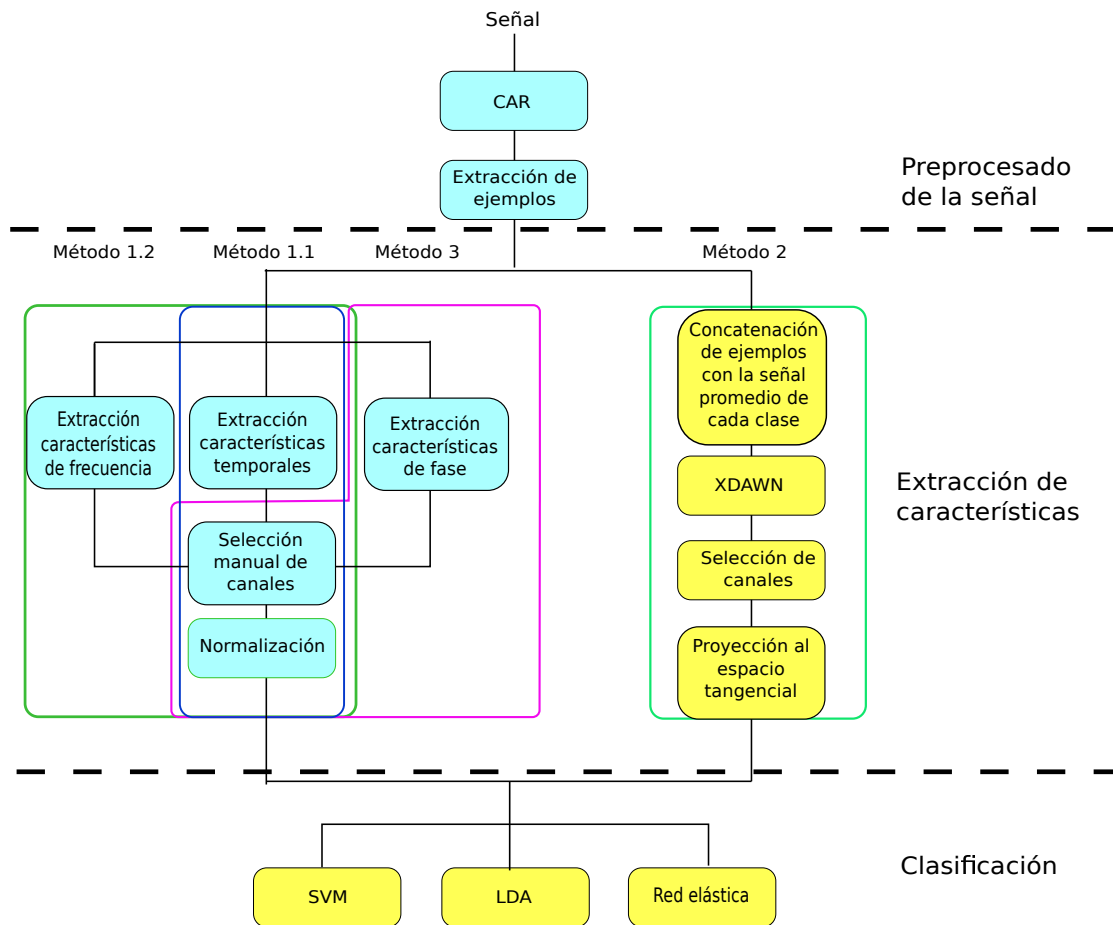


Figura 4.3: Se muestra el proceso completo de clasificación en el que se observa el preprocesado de la señal, la extracción de características y los clasificadores utilizados. Las cajas de color azul indican los algoritmos que han sido implementados y las de color naranja, los algoritmos que han sido utilizados con la ayuda de librerías externas.

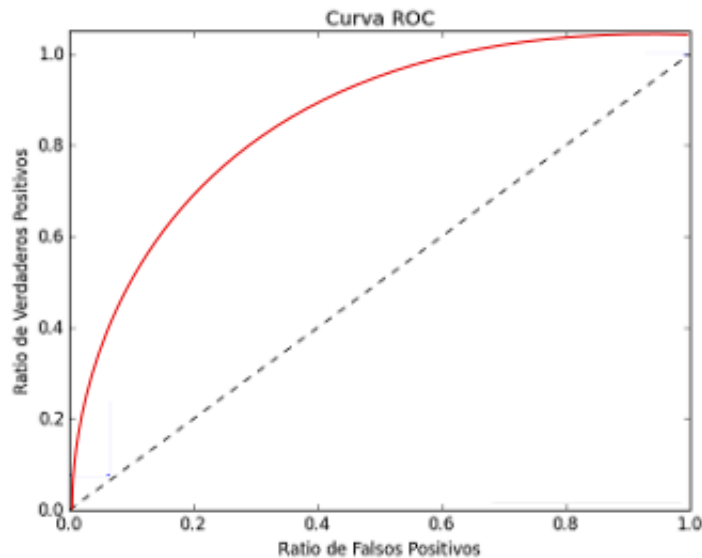


Figura 4.4: Se muestra un ejemplo de curva ROC de la que se obtiene el AUC.

4.3. Métricas de comparación

Para evaluar los resultados de clasificación se han empleado dos métricas, la matriz de confusión para un umbral fijo, y el área bajo la curva (AUC). La matriz de confusión es una tabla que permite visualizar el rendimiento de un clasificador, en la que cada columna representa el número de predicciones de cada clase, y cada fila las instancias reales de la clase. El área bajo la curva es una métrica de evaluación que se calcula utilizando el área bajo la curva ROC [30]. La curva ROC se obtiene como la razón entre los verdaderos positivos y los falsos positivos obtenidos al clasificar, para todos los posibles umbrales de clasificación, que se encuentran en el rango $[0,1]$.

4.4. Herramientas e implementación

El foco principal del trabajo es la realización de una comparativa entre diferentes procesos para clasificar trials. Con este objetivo, el trabajo se ha dividido en 3 partes.

En la primera parte del trabajo, se ha estudiado la bibliografía correspondiente al estado del arte en la clasificación de BCIs y se ha buscado implementaciones de los algoritmos que componen los procesos de clasificación. Por un lado, se disponía del código correspondiente de la metodología de la universidad de Zaragoza implementado en Matlab. Por otro lado, se encontraron funciones útiles para trabajar con la geometría Riemanniana implementadas en Python. Esto era un problema a la hora de obtener los resultados, ya que había que utilizar dos entornos diferentes para programar, por lo que se decidió unificar todo el código en un solo lenguaje de programación. Tras realizar un análisis del coste que supondría migrar un código u otro, se eligió migrar el código

de Matlab a Python, por lo que se migró todo el código correspondiente a la lectura de ficheros que contenían la información de la señal, extracción y etiquetado de trials, preprocesado, y extracción de características. Para asegurarse de que la migración se realizó correctamente, se realizó una validación de los resultados obtenidos en Python con respecto a los obtenidos anteriormente en Matlab.

En la segunda parte, se añadieron las características de fase al proyecto cuya extracción ha sido implementada en su totalidad junto al submuestreo, concatenación y normalización de los trials. Además, se han reutilizado los filtros de paso banda y CAR, implementados en el preprocesado de la parte anterior, y la implementación de los clasificadores proporcionada por la librería scikit-learn [34], que se han utilizado en las 3 metodologías comparadas.

En la parte final, se han creado las funciones necesarias para la comparativa. Sin embargo, estas funciones requerían de la elección de muchos parámetros para conseguir los mejores resultados posibles. Debido a esto, había que modificar el código cada vez, lo que facilitaba la introducción de errores en el proceso, por lo que se decidió realizar una secuencia de procesamiento genérica que permite añadir o eliminar algoritmos y cambiar sus parámetros modificando un único fichero en formato yaml. Esto además permite que otras personas puedan reutilizar el código sin tener que manipularlo directamente y facilita su mantenimiento, lo que es un valor añadido. Este fichero consta de dos secciones, la primera sección es donde se incluyen los archivos importados necesarios para ejecutar los métodos, y en la segunda se introducen los métodos junto a sus parámetros necesarios. Este comportamiento ha sido implementando utilizando la función Pipeline de la librería sklearn [32]. El fichero tiene la siguiente forma:

```
imports :
  sklearn.linear_model :
    - ElasticNet

  classif :
    - XdawnCovariances
    - TangentSpace
    - ElectrodeSelect

pipeline :
  - XdawnCovariances :
    nfilter : 5
    subelec : range(0,60,1)
  - ElectrodeSelect :
    nfilters : 5
    nelec : 10
    metric : "'riemann'"
  - TangentSpace :
    metric : "'logeuclid'"
```

```
    tupdate: False
- ElasticNet:
    l1_ratio: 0.5
    alpha: 2e-4
    normalize: True
```

Para la metodología del concurso, se ha utilizado la librería PyRiemann para implementar la selección de canales y proyección al espacio tangencial, junto al algoritmo XDAWN, que es implementado aplicando la optimización vista en [33]. Por último, se han implementado los algoritmos de validación cruzada utilizados para la elección de parámetros, junto las funciones que devuelven los resultados en un fichero Excel para facilitar el análisis posterior de los datos.

5. Resultados de Clasificación

5.1. Clasificación por sujeto

En este apartado se presentan los resultados de clasificación por sujeto, en los que se separan los datos de cada sujeto en datos de entrenamiento de test. En primer lugar, y debido a que hay muchas combinaciones posibles entre los sujetos que realizaron el experimento (12 sujetos), clasificadores (SVM, LDA, Red Elástica), conjuntos de eventos a detectar (Inicios, Desvíos) y metodologías de procesamiento (Metodología 1.1, 1.2; Metodología 2; Metodología 3), en la Figura 5.1 se muestran los resultados obtenidos de promediar los porcentajes de clasificación y las AUC para el conjunto de todos los sujetos. En dicha gráfica se puede apreciar que los resultados obtenidos utilizando el clasificador SVM son consistentemente superiores a los otros dos clasificadores (LDA y Red Elástica) obteniendo porcentajes de clasificación un 7% superior en cuanto a la detección de Inicios y un 3% respecto a los desvíos. Por esta razón, en lo correspondiente a los próximos análisis vamos a centrarnos únicamente en los resultados obtenidos por el clasificador SVM y los resultados conseguidos por LDA y EN podrán ser consultados en el Anexo A.

La Tabla 5.1 muestra los resultados de la clasificación de los eventos de desvío del movimiento. En este caso, se obtienen unos resultados similares, siendo la Metodología 2 (Metodología del concurso), con un AUC del 76.8% y una tasa de acierto media del 71.9%, la que mejores resultados presenta, seguida de la Metodología 1.1 (Metodología

Tabla 5.1: Resultados de la clasificación por sujeto para los eventos de desvío del movimiento.

Desvíos		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.651	0.359	0.746
	Erróneos	0.275	0.725	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.598	0.402	0.743
	Erróneos	0.27	0.73	
Metodología 2 (Concurso)	Correctos	0.682	0.318	0.768
	Erróneos	0.244	0.756	
Metodología 3 (Fase)	Correctos	0.645	0.355	0.743
	Erróneos	0.302	0.698	

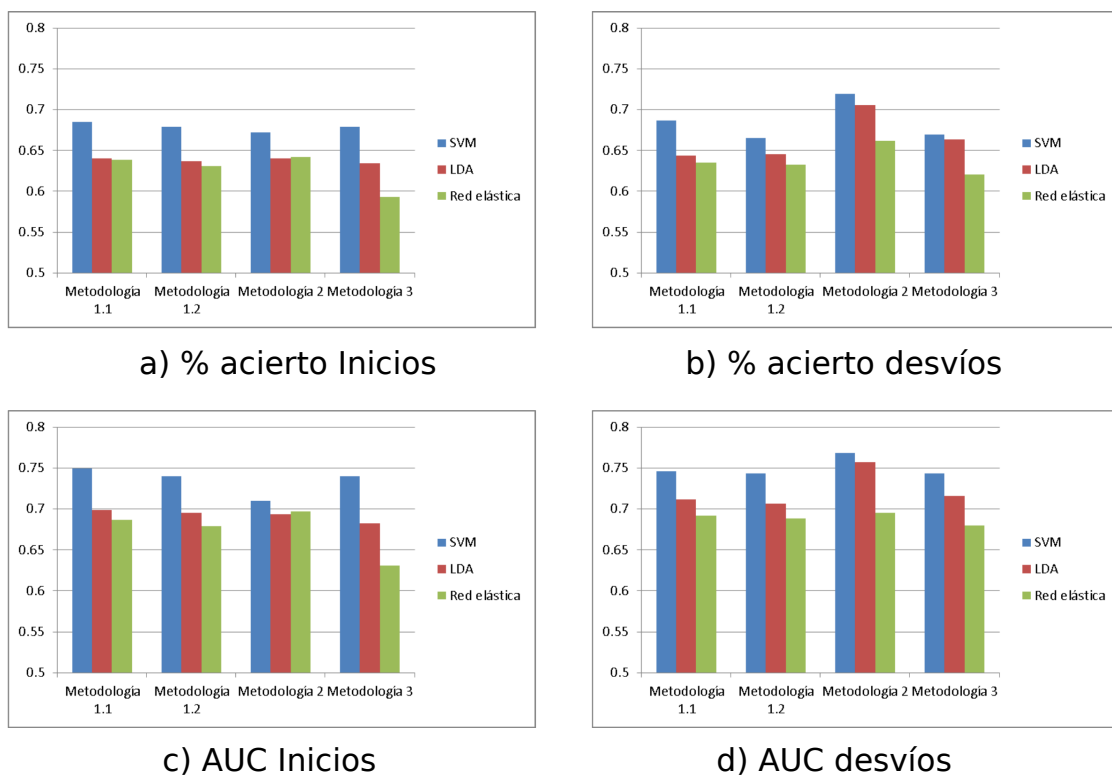


Figura 5.1: Se muestran los resultados de clasificación medios de cada metodología por clasificador.

Tabla 5.2: Resultados de la clasificación por sujeto para los eventos de inicio del movimiento.

Inicios		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.686	0.314	0.749
	Erróneos	0.324	0.676	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.697	0.303	0.739
	Erróneos	0.34	0.66	
Metodología 2 (Concurso)	Correctos	0.666	0.334	0.71
	Erróneos	0.323	0.677	
Metodología 3 (Fase)	Correctos	0.695	0.305	0.74
	Erróneos	0.335	0.665	

de la universidad con características temporales), con un AUC del 74.6% y tasa de acierto del 68.7%. En tercer lugar se encuentra la Metodología 3 (Metodología con características de fase), con un AUC del 74.3% y tasa de acierto del 67.2% y en último lugar, la Metodología 1.2 (Metodología de la universidad con características temporales y de frecuencia), con un AUC del 74.3% y tasa de acierto del 66.4%. Los eventos de error presentan una tasa de acierto mayor en su clasificación respecto a los de la clase de no error de entre un 5 y un 13%. Esto se puede producir porque el modelo tiene cierto sesgo hacia la clase de error provocando que cada trial se detecte como trial erróneo más fácilmente que como trial correcto. En la Figura 5.2a y 5.2c se muestran los resultados con el clasificador SVM de los desvíos de cada sujeto, en los que se puede observar que el Sujeto 3 presenta una tasa de acierto de menos del 45% con las metodologías de la universidad y de la fase, mientras que la metodología del concurso obtiene unos resultados más uniformes para el conjunto de sujetos sin bajar del 60% en ninguno, por lo que esta última es más robusta.

Los resultados de los eventos de inicio de movimiento están representados en la Tabla 5.2. Las metodologías de la universidad y de características de fase tienen unos resultados muy parejos entre sí con un AUC entre 73.9 y 74.9% y una tasa de acierto del 68%, mientras que la metodología del concurso obtiene unos resultados ligeramente peores con un AUC del 71% y tasa de acierto del 67.2%. Las clases de error y de no error presentan una tasa de acierto en su clasificación prácticamente idéntica. Se pueden ver los resultados por sujeto con el clasificador SVM de los eventos de inicio del movimiento en la Figura 5.2b y 5.2d. En este caso hay poca variación en los resultados entre sujetos y métodos.

5.2. Clasificación por transferencia entre sujetos

En este apartado se presentan los resultados de clasificación por transferencia entre sujeto en los que se usan 9 sujetos para entrenar y 3 para test. Una vez más, comparando los resultados obtenidos para los 3 distintos clasificadores (Figura 5.3), SVM consigue los mejores resultados, obteniendo un promedio de 4% de detección de aciertos mejor que

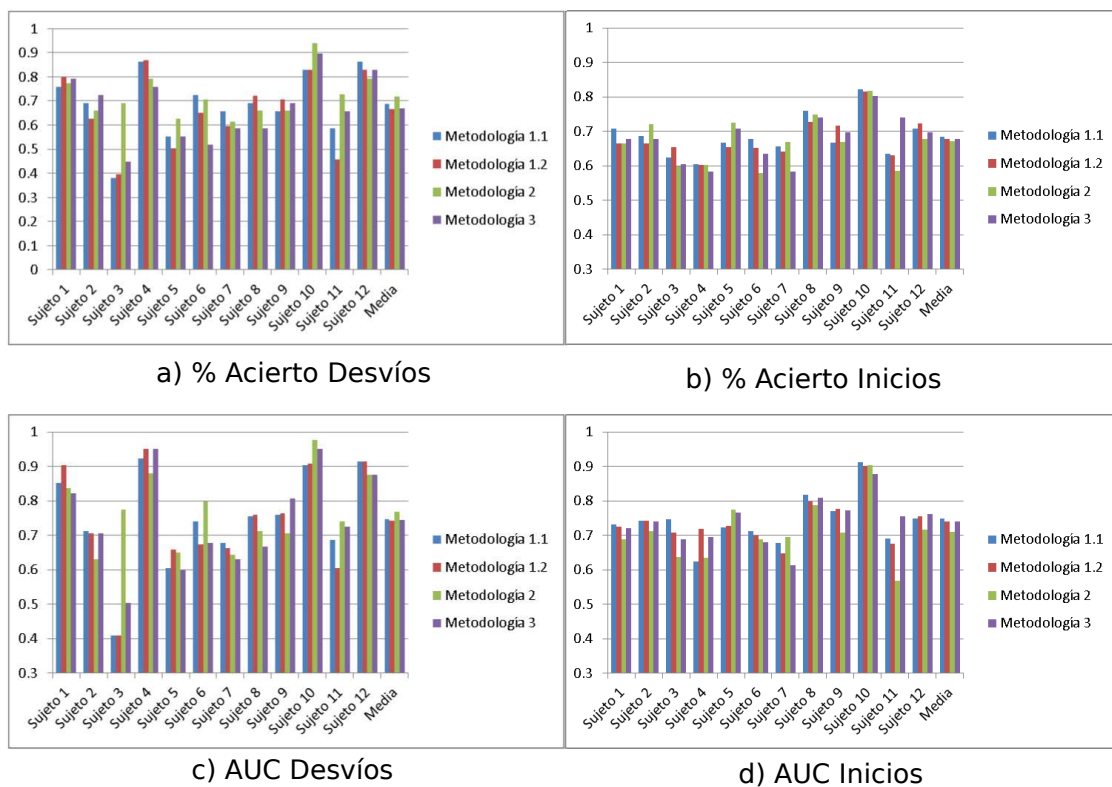
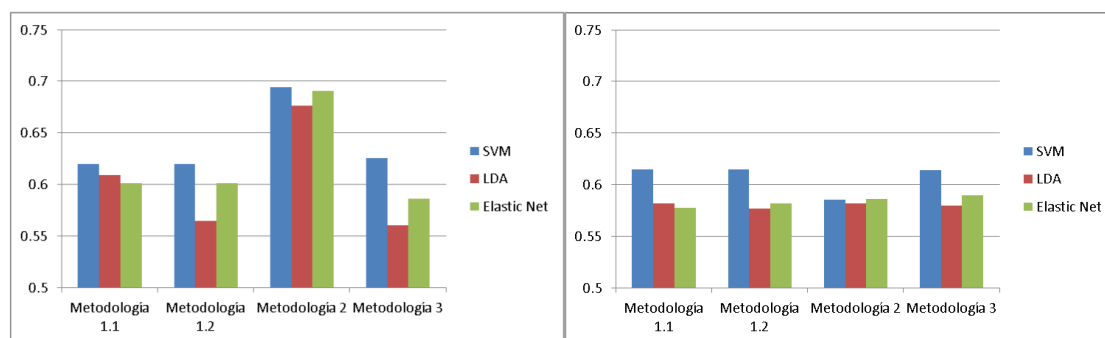
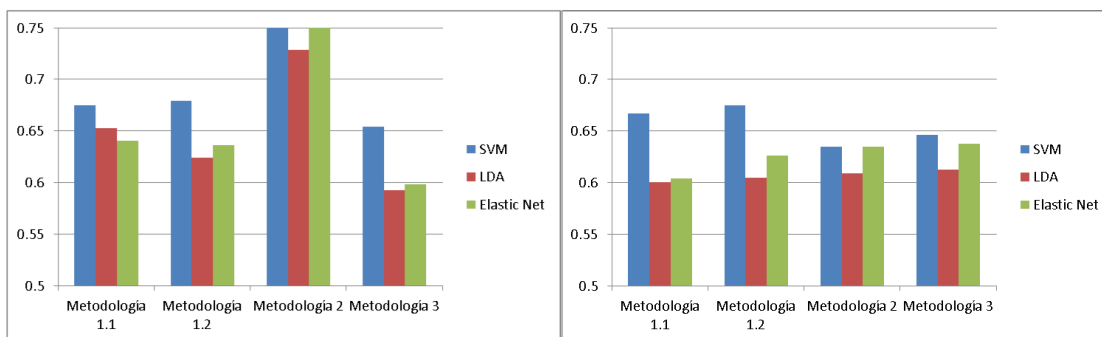


Figura 5.2: Se muestran los resultados de cada sujeto por metodología utilizando el clasificador SVM para la clasificación.



a) Tasa de acierto en Desvíos

b) Tasa de acierto en Inicios



c) AUC en Desvíos

d) AUC en Inicios

Figura 5.3: Se muestran los resultados de la clasificación por transferencia entre sujetos.

los otros clasificadores. Por ello, los próximos análisis únicamente reportan los resultados obtenidos por el clasificador SVM y los resultados obtenidos con LDA y Red elástica pueden ser consultados en el Anexo A.

En la Tabla 5.3, se muestran los resultados para los eventos de desvío del movimiento de la clasificación por transferencia entre sujetos. Los mejores resultados se obtienen con la metodología 2 con una clara mejoría sobre el resto, en la que se obtiene una tasa de acierto del 69.3 %, que supone una mejora de entre el 6.5 y el 9 % respecto al resto de metodologías, y un AUC del 76 %, que es un incremento de entre el 8 y el 10 % respecto al resto, que presentan unos resultados parecidos entre sí. Los eventos erróneos se clasifican con una mayor fiabilidad que los correctos llegando hasta 83 % de acierto en el caso de la metodología del concurso, mientras que los eventos no erróneos se clasifican peor, llegando al 61 % de tasa de acierto mayor. Una posible explicación de este efecto es que el modelo esté sesgado hacia la clase de error provocando que cada trial se detecte como trial erróneo más fácilmente que como trial correcto.

Los resultados de los eventos de inicio del movimiento se recogen en la Tabla 5.4. Se obtienen unos resultados muy parecidos entre las metodologías, siendo la metodología 1.2 la que obtiene el mejor AUC con un 67.8 %, ya la metodología 1.1 la que obtiene la mejor

Tabla 5.3: Resultados de la clasificación por transferencia entre sujetos para los eventos de desvío del movimiento.

Desvíos		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.452	0.548	0.675
	Erróneos	0.212	0.788	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.456	0.544	0.679
	Erróneos	0.207	0.793	
Metodología 2 (Concurso)	Correctos	0.552	0.448	0.76
	Erróneos	0.165	0.835	
Metodología 3 (Fase)	Correctos	0.612	0.388	0.654
	Erróneos	0.362	0.638	

Tabla 5.4: Resultados de la clasificación por transferencia entre sujetos para los eventos de inicio del movimiento.

Inicios		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.527	0.473	0.662
	Erróneos	0.245	0.755	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.548	0.452	0.678
	Erróneos	0.298	0.702	
Metodología 2 (Concurso)	Correctos	0.545	0.455	0.642
	Erróneos	0.324	0.676	
Metodología 3 (Fase)	Correctos	0.509	0.491	0.651
	Erróneos	0.282	0.718	

tasa de acierto con un 64.1 %. La metodología 3 tiene unos resultados ligeramente peores que las metodologías anteriores, con un AUC del 65.1 % y tasa de acierto del 61.3 %, aunque un poco mejores que la metodología 2, que obtiene resultados casi idénticos con un AUC del 64.3 % y tasa de acierto del 61 %. Las clases de error se clasifican con una fiabilidad de entre el 67 y el 75 %, mientras que los eventos de no error, con una fiabilidad de entre el 50 y el 55 %.

6. Conclusiones

Las BCI proporcionan a las personas con discapacidad una forma de mejorar su calidad de vida. Estas interfaces son propensas al error al identificar la intención de usuario, por lo que se suelen utilizar los potenciales de error para ayudar a la identificación. Sin embargo, estos potenciales son difíciles de detectar debido al ruido de fondo que se introduce al grabar la actividad cerebral del sujeto, por lo que se han desarrollado diferentes metodologías para detectar estos potenciales lo mejor posible.

En este trabajo se ha realizado una comparativa de diferentes metodologías y clasificadores, utilizando los datos de un experimento en el que se grababa 2 conjuntos de datos de 2 tipos de eventos diferentes (eventos de desvío y eventos de inicio), en clasificación por transferencia entre sujetos y clasificación por sujeto. De los clasificadores utilizados, el que mejor resultado ha dado ha sido el clasificador SVM, que ha obtenido la mejor clasificación en casi todos los casos.

Por un lado, se ha utilizado una metodología empleada por la universidad con características temporales, y una combinación de características temporales y frecuenciales. En los resultados no se observa mucha diferencia en la clasificación. En la clasificación por transferencia entre sujetos los resultados decaen bastante como era de esperar al utilizar este método, con un decremento de casi el 10% respecto a la clasificación por sujeto, que conseguía un acierto en la clasificación de en torno al 70-75%.

Por otro lado, se ha utilizado una metodología sacada de un concurso en el que se proponía un método que hacía uso de la Geometría Riemanniana que tiene unas propiedades que permiten obtener una buena generalización entre sujetos. Los resultados obtenidos, varían bastante dependiendo del tipo de evento, en los eventos de desvío se obtienen los mejores resultados tanto en la clasificación por transferencia entre sujetos como por sujeto, obteniendo un AUC del 76% en ambos casos. Sin embargo, en los eventos de inicio se obtienen peores resultados que con la metodología de la universidad tanto para clasificación por sujeto como por transferencia entre sujetos, con una degradación del 3% en ambos casos. Además, esta metodología es más costosa en tiempo que la de la universidad al por transferencia entre sujetos la selección de parámetros mediante validación cruzada y el cálculo de la media geométrica que requiere mucho tiempo de computo en comparación al resto del proceso.

Finalmente, se ha utilizado una metodología que hace uso de las características de fase. Esta última metodología obtiene resultados muy parecidos a la metodología de la universidad aunque ligeramente peores.

6 Conclusiones

Estos resultados no son todo lo buenos que se hubiera querido y habría que probar ambos métodos en más sujetos y tareas para llegar a unas mejores conclusiones. La detección de ERPs es un tema complejo por el que aún queda mucho por descubrir. No obstante, se han podido aprender y utilizar diferentes técnicas que pueden servir de base para conseguir mejores resultados.

ANEXOS



Anexos A

Resultados adicionales

En este anexo se presenta los resultados obtenidos con los clasificadores LDA y Red Elástica.

A.1. Resultados LDA por sujeto

A continuación se muestran los resultados obtenidos con el LDA.

Tabla A.1: Resultados de la clasificación por sujeto para los eventos de desvío del movimiento.

Desvíos		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.656	0.344	0.711
	Erróneos	0.372	0.628	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.656	0.344	0.706
	Erróneos	0.366	0.634	
Metodología 2 (Concurso)	Correctos	0.661	0.339	0.757
	Erróneos	0.25	0.75	
Metodología 3 (Fase)	Correctos	0.692	0.308	0.716
	Erróneos	0.372	0.628	

Tabla A.2: Resultados de la clasificación por sujeto para los eventos de inicio del movimiento.

Inicios		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.655	0.345	0.699
	Erróneos	0.371	0.629	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.668	0.332	0.695
	Erróneos	0.395	0.605	
Metodología 2 (Concurso)	Correctos	0.625	0.375	0.693
	Erróneos	0.356	0.654	
Metodología 3 (Fase)	Correctos	0.641	0.359	0.682
	Erróneos	0.371	0.629	

A.2. Resultados Red Elástica por sujeto

A continuación se muestran los resultados obtenidos con la Red Elástica.

Tabla A.3: Resultados de la clasificación por sujeto para los eventos de desvío del movimiento.

Desvíos		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.671	0.329	0.691
	Erróneos	0.41	0.59	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.656	0.344	0.688
	Erróneos	0.392	0.608	
Metodología 2 (Concurso)	Correctos	0.598	0.402	0.695
	Erróneos	0.276	0.724	
Metodología 3 (Fase)	Correctos	0.646	0.354	0.679
	Erróneos	0.41	0.59	

Tabla A.4: Resultados de la clasificación por sujeto para los eventos de inicio del movimiento.

Inicios		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.645	0.355	0.686
	Erróneos	0.367	0.633	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.651	0.349	0.678
	Erróneos	0.39	0.61	
Metodología 2 (Concurso)	Correctos	0.595	0.405	0.696
	Erróneos	0.312	0.688	
Metodología 3 (Fase)	Correctos	0.598	0.402	0.63
	Erróneos	0.412	0.588	

[H]

A.3. Resultados LDA por transferencia entre sujetos

A continuación se muestran los resultados obtenidos con el LDA.

Tabla A.5: Resultados de la clasificación por transferencia entre sujetos para los eventos de desvío del movimiento.

Desvíos		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.543	0.457	0.652
	Erróneos	0.324	0.676	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.52	0.48	0.624
	Erróneos	0.385	0.615	
Metodología 2 (Concurso)	Correctos	0.634	0.366	0.728
	Erróneos	0.292	0.708	
Metodología 3 (Fase)	Correctos	0.525	0.475	0.592
	Erróneos	0.401	0.599	

Tabla A.6: Resultados de la clasificación por transferencia entre sujetos para los eventos de inicio del movimiento.

Inicios		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.51	0.49	0.61
	Erróneos	0.317	0.683	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.506	0.494	0.625
	Erróneos	0.346	0.654	
Metodología 2 (Concurso)	Correctos	0.522	0.461	0.606
	Erróneos	0.375	0.625	
Metodología 1.3 (Fase)	Correctos	0.525	0.475	0.585
	Erróneos	0.422	0.578	

A.4. Resultados Red Elástica por transferencia entre sujetos

A continuación se muestran los resultados obtenidos con la Red Elástica.

A Resultados adicionales A.4 Resultados Red Elástica por transferencia entre sujetos

Tabla A.7: Resultados de la clasificación por transferencia entre sujetos para los eventos de desvío del movimiento.

Desvíos		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.451	0.549	0.64
	Erróneos	0.297	0.703	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.484	0.516	0.636
	Erróneos	0.292	0.708	
Metodología 2 (Concurso)	Correctos	0.58	0.42	0.761
	Erróneos	0.198	0.802	
Metodología 3 (Fase)	Correctos	0.57	0.43	0.598
	Erróneos	0.4	0.6	

Tabla A.8: Resultados de la clasificación por transferencia entre sujetos para los eventos de inicio del movimiento.

Inicios		Correctos	Erróneos	AUC
Metodología 1.1 (Temporal)	Correctos	0.498	0.502	0.611
	Erróneos	0.382	0.618	
Metodología 1.2 (Temporal + Frecuencia)	Correctos	0.508	0.492	0.619
	Erróneos	0.377	0.623	
Metodología 2 (Concurso)	Correctos	0.512	0.488	0.612
	Erróneos	0.336	0.664	
Metodología 3 (Fase)	Correctos	0.521	0.379	0.617
	Erróneos	0.366	0.634	

Bibliografía

- [1] J R Millán, R Rupp, G Müller-Putz, R Murray-Smith, C Giugliemma, M Tangermann, C Vidaurre, F Cincotti, A Kübler, R Leeb, et al. Combining brain–computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers in neuroscience*, 4, 2010.
- [2] A Schlögl, F Lee, H Bischof, and Gt Pfurtscheller. Characterization of four-class motor imagery eeg data for the bci-competition 2005. *Journal of neural engineering*, 2(4):L14, 2005.
- [3] P Ferrez and J Millán. Simultaneous real-time detection of motor imagery and error-related potentials for improved bci accuracy. In *Proceedings of the 4th international brain-computer interface workshop and training course*, number LIDIAP-CONF-2008-019, 2008.
- [4] I Iturrate, L Montesano, and J Minguez. Task-dependent signal variations in eeg error-related potentials for brain computer interfaces. *Journal of neural engineering*, 10(2):026024, 2013.
- [5] Zhang and X Wang. Temporal and frequency feature extraction with canonical variates analysis for multi-class imagery task. In *Control and Decision Conference, 2008. CCDC 2008. Chinese*, pages 2228–2232. IEEE, 2008.
- [6] B. Rivet, V. Souiloumiac, A. and Attina, and G. Gibert. x dawn algorithm to enhance evoked potentials: Application to brain-computer interface. *IEEE Transactions on Biomedical Engineering*, 56(8):2035, August 2009.
- [7] K Su and K Robbins. Subject specific parameter selection for the eeg classifier using common spatial patterns. In *Proceedings of the 2012 ACM Research in Applied Computation Symposium*, pages 68–69. ACM, 2012.
- [8] A Subasi and M Gursoy. Eeg signal classification using pca, ica, lda and support vector machines. *Expert Systems with Applications*, 37(12):8659–8666, 2010.
- [9] I Güler and E Beyli. Multiclass support vector machines for eeg-signals classification. *Information Technology in Biomedicine, IEEE Transactions on*, 11(2):117–126, 2007.

- [10] R Kottaimalai, M Rajasekaran, V Selvam, and B Kannapiran. Eeg signal classification using principal component analysis with neural network in brain computer interface applications. In *Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN), 2013 International Conference on*, pages 227–231. IEEE, 2013.
- [11] F Lotte, M Congedo, A Lécuyer, and F Lamarche. A review of classification algorithms for eeg-based brain–computer interfaces. *Journal of neural engineering*, 4, 2007.
- [12] Bci challenge 2015. <https://www.kaggle.com/c/inria-bci-challenge>. Accessed: 2015-11-10.
- [13] M De Vos, M Kroesen, R Emkes, and S Debener. P300 speller bci with a mobile eeg system: comparison to a traditional amplifier. *Journal of neural engineering*, 11(3):036008, 2014.
- [14] E Niedermeyer and FHL da Silva. *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins, 2005.
- [15] M Falkenstein, J Hoormann, S Christ, and J Hohnsbein. Erp components on reaction errors and their functional significance: a tutorial. *Biological psychology*, 51(2):87–107, 2000.
- [16] HT van Schie, RB Mars, MGH Coles, and H Bekkering. Modulation of activity in medial frontal and motor cortices during error observation. *Nature neuroscience*, 7(5):549–554, 2004.
- [17] PW Ferrez. Error-related eeg potentials generated during simulated brain–computer interaction. *Biomedical Engineering, IEEE Transactions on*, 55(3):923–929, 2008.
- [18] C Carter, T Braver, D Barch, M Botvinick, D Noll, and J Cohen. Anterior cingulate cortex, error detection, and the online monitoring of performance. *Science*, 280(5364):747–749, 1998.
- [19] A Schlögl, M Slater, and G Pfurtscheller. Presence research and eeg. In *Proceedings of the 5th International Workshop on Presence*, pages 9–11, 2002.
- [20] A Barachant, S Bonnet, M Congedo, and C Jutten. Bci Signal Classification using a Riemannian-based kernel. pages 97–102, April 2012.
- [21] M Congedo, A Barachant, and A Andreev. A new generation of brain-computer interface based on riemannian geometry. *arXiv preprint arXiv:1310.8115*, 2013.
- [22] D J McFarland, L M McCane, S V David, and J R Wolpaw. Spatial filter selection for EEG-based communication Electroencephalogr. *Clin. Neurophysiol*, 103, 1997.
- [23] P Welch. The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. 5, 1967.

- [24] M Cohen. Error-related medial frontal theta activity predicts cingulate-related structural connectivity. *NeuroImage*, 55(3):1373–1383, 2011.
- [25] A Barachant and S Bonnet. Channel selection procedure using riemannian distance for BCI applications. In *2011 5th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 348–351. 2011.
- [26] O. Ledoit and M. Wolf. A well-conditioned estimator for large- dimensional covariance matrices. *Journal of multivariate analysis*, 88:365–411, 2004.
- [27] I. Iturrate, L. Montesano, and J. Minguez. Single trial recognition of error-related potentials during observation of robot operation. In *Proc of the Annual Int Conf of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2010.
- [28] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. A review of classification algorithms for EEG-based brain-computer interfaces. *J Neural Eng*, 4(2):R1–R13, June 2007.
- [29] T Tian and Z Li. A spatio-temporal solution for the eeg/meg inverse problem using group penalization methods. *Statistics and its Interface*, 4(4):521–533, 2011.
- [30] A Stewart, A Nuthmann, and G Sanguinetti. Single-trial classification of eeg in a visual object task using ica and machine learning. *Journal of neuroscience methods*, 228:1–14, 2014.
- [31] JM Phillips, RS Ehrlichman, and SJ Siegel. Mecamylamine blocks nicotine-induced enhancement of the p20 auditory event-related potential and evoked gamma. *Neuroscience*, 144(4):1314–1323, 2007.
- [32] Pipeline. <http://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html>. Accessed: 2015-11-10.
- [33] B Rivet, H Cecotti, A Souloumiac, El Maby, and J Mattout. *Theoretical analysis of xDAWN algorithm: application to an efficient sensor selection in a P300 BCI*. 19th European Signal Processing Conference (EUSIPCO 2011), August 2011.
- [34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.